merit to a professional musician. However, this was not achieved without an appreciable amount of experimentation. The miscellaneous parameters appearing in the rules of composition seemed to affect the results in a rather erratic way, and it was not easy to adjust them all by trial and error in order to produce acceptable results. Perhaps, in any future work of this kind, the idea of negative feedback should be used to control some of these parameters as well as the rate of composition. However, there would obviously be a risk of instability if this were carried too far.

The composition of music by means of a computer introduces yet another new complication in the already turbulent world of modern music. The author's experience has convinced him that the computer program is hardly a substitute for the human composer but is rather a new (and somewhat devious) medium of expression. Composers have already faced the challenge of expressing their work in the form of, e.g., schedules for the copying of sections of tape recordings through filters (in composing musique concrète), or punched tape to control electronic tone generators. Now they may also express themselves in the form of computer programs (or in the form of controlling parameters to be supplied to music-generating programs).

There is, however, one significant difference between composition by computing and by orthodox methods, namely that when using a computer the composer is much further removed from the final result (i.e. the music) than he is when writing an ordinary score. Experience will no doubt help a great deal, but it is doubtful whether composers will ever be able to foresee very clearly the full result of every choice that they make when feeding the computer.

Perhaps in the end we shall see musical composition taking the form of a co-operative venture between the human composer and the computer, with the computer supplying a number of plausible passages along lines suggested by the composer, who in turn selects the ones he wants and calls for further variations and ornamentations as required. This will, of course, call for suitably designed input and output devices. It would probably be done most effectively by means of a time-sharing program in a very powerful computer, since it would require spasmodic bursts of rapid computing.

Acknowledgements

The author is indebted to the British Broadcasting Corporation for their help and encouragement in this project, and in particular to Mr. Lionel Salter who acted as musical adviser, translated and arranged the computer output, and also made valuable comments on this paper. He also wishes to thank Ferranti Limited for the use of Pegasus and for permission to publish this paper.

Reference

HILLER, L. A., and ISAACSON, L. M. (1959). *Experimental Music*, New York: McGraw-Hill.

Correspondence

To the Editor,
The Computer Journal.
Sir,

"*The LL^T and QR methods for symmetric tridiagonal matrices*"

In their paper, "The LL^T and QR methods for symmetric tridiagonal matrices" (this *Journal*, Vol. 6, p. 99), James M. Ortega and H. F. Kaiser have successfully eliminated all square roots from the formulae for the LL^T-transformation (symmetric or "Choleski" modification of the LR-transformation) if applied to a symmetric tridiagonal matrix, and derive the formulae

$$\left.\begin{aligned} q_1 &= a_1 \\ e_i &= b_i^2/q_i \\ \bar{a}_i &= q_i + e_i \\ q_{i+1} &= a_{i+1} - e_i \\ b_1^2 &= q_{i+1}e_i \\ \bar{a}_n &= q_n \end{aligned}\right\} \text{ for } i = 1, 2, \ldots, n-1$$

(q and e stand for the authors' d^2 and s^2 respectively).

This would seem to be a great achievement, but by comparison with the basic formulae of the quotient-difference algorithm (Rutishauser, 1957; Henrici 1958),

$$\left.\begin{aligned} \bar{q}_1 &= q_1 + e_1 \\ \bar{e}_i &= q_{i+1}e_i/\bar{q}_i \\ \bar{q}_{i+1} &= q_{i+1} + e_{i+1} - \bar{e}_i \end{aligned}\right\} \begin{array}{l}\text{for } i = 1, 2, \ldots, n-1, \text{ with} \\ e_n \text{ taken to be zero,}\end{array}$$

it becomes obvious that the authors have nothing but re-established the quotient difference algorithm which since long has been used to compute eigenvalues of symmetric tridiagonal matrices, and which, incidentally, was the starting point from which the LR-transformation was derived by generalization.

The abbreviated QR-transformation on the other hand seems to be new and useful. However, in order to improve its numerical stability, it would be advantageous to carry not only the s_i^2, but also the corresponding $c_i^2 = 1 - s_i^2$ in the calculation.

Yours faithfully,
H. RUTISHAUSER.

Swiss Federal Institute of Technology,
Zurich, Switzerland.
13 May, 1963

RUTISHAUSER, H. (1957). *Der Quotienten-Differenzen Algorithmus*, Mitt. Inst. angew. Math. der ETH, Nr. 7, Birkhäuser Verlag, Basel.

HENRICI, P. (1958). *The Quotient-Difference Algorithm*, Nat. Bur. Standards, Applied Math. Series Vol. 49, p. 23–46.